Feedback Comments
: "Good job Vince! Looks great!",
: "Good job Will! I really like the design and I found it intuitive to use!",
: "Awesome job Davon! Really funny concept!!!",
: "Really good job Mike! I like the clean look to it. Things are presented well so that I know how to use the program! ",
: "Looks great Jason. I like the look of it and it's very intuitive. I added \"Dan Burger\" twice and there were 2 separate entries for both. Although not a requirement, I would check to see if the name was already entered and alert the user about it",
: "Hey Angel, I checked your herokuapp and noticed that it's running into an application error, so I cloned your repo, looked around and deployed for myself to see what's up.\n\nIt looks like you've got the code you needed for deployment of your mysql server (assuming you've added the JAWSDB add-on on Heroku). But you've got the if/else switch commented out in favor of the local-only version. You should review the README.md that Albert made (Pavan posted) on how to deploy to Heroku with a mySQL db. If you have any questions after that feel free to reach out.\n\nAside from the Deployment problems (which are paramount), I noticed a few instances of missing semi-colons. Be mindful of these as it's just proper convention.\n\nAlso in terms of the actual application of the App. There's a strange bug that adds a burger to both columns, when a user adds a \"Devoured!\" burger. I believe the original flow idea was to add a burger and have it show up on a column of \"not devoured\" burgers. Then use a \"Devour It\" button to move it to the other column. The bonus would've been another action which would delete the burger from the Devoured Column.\n\nOverall, you've got the actions down, but take a minute and think of how the app should work and you'll find the bug I was talking about.\n\nAlso, I briefly checked the responsiveness of the site itself, it looks like there's still a hard coded width of about 960px, which makes the page have to scroll more on horizontal. Don't forget about BootStrap with it's column views, or if you want a really barebones way of learning how to use a responsive grid, look up Skeleton Grid.\n\nAlso, there's no need to include a `js` with the instructions. You're better off writing a README.md for other people to see and you can remind yourself what your next step will be if you make it detailed enough.",

: "Hey David, I cloned your app and tried to run it, but I ran into a few bugs that had to be squashed.\n\nThe first thing that needed to be changed was the port that you create your mySQLconncetion on. You've specifed it to be port 3000, but we're already using port 3000 for you node server. By default, mysql port is usually `3306`, it's best not to change that unless you need to run another server for whatever reason.\n\nAfter that I was running into a missing partial. I'm not sure why you have the handlebars for including a partial, but don't have the `burgers/partial`in your views. It looks like your partials are also not rending the burger_names, since in your {{#each}} statements you only refer to it as `this.name` instead of `this.burger_name`\n\n\nDid you get a chance to test this before submitting? I see you were able to submit early, and it was close to working, but the few mistakes were breaking the app completely.\n\n",
: "Hey David, I took a look at your deployed app and reviewed your code on GitHub.\n\nVery nice job, you've nailed the functionality of the app. Aside from a missing `;` in your `orm.js` and your indentation on your `index.handlebars, your syntax is looking really good. It helps clearly read what you're doing without having to think too much about it.\n\nGreat job on the functionality, but your styling also needs work. While you can lean more towards backend or frontend, you're building up to a full-stack dev, so don't forget to polish up your front-end skills. I think you were looking for maybe floats on those devoured and non devoured list (which btw also don't have headers). Don't forget the box model. When I forget things, I either look them up, hope bootstrap/materialize will take care of it for me, or open up js.fiddle, back some divs with background colors and specified heights, and just play around. e.g. floating left/right, nesting-divs and messing with margins and padding.\n\n\nOverall great job, polish up the front-end and maybe fill out that README.md a bit more. Also don't forget about your `.gitignore`, no point in having all those extra npm modules taking up space.",
: "Hey Jimmy, I cloned your Burger App and took a look around...\n\nVery nice job on the functionality and getting it to work with Sequelize. The ```Burger.sync()``` and ```console.log(Burger)``` was a nice touch. \n\nIt looks like you've got this app setup for deployment on the latest commit. But why use a `selectedSource = source.whatever`? You can build it as an if else around `process.env.Jawsdburl`, so your code works whether it's local or deployed. \n\nIt looks like you've nailed the functionality of the homework which is great. You may want to tweak the front-end when you get time, the layout seems a little sparse (by nature it is for the app), but I noticed a buch of `<br>` tags. Why bother with that when you can control the spacing with `margin` and `padding`?\n\nLastly, don't forget a .gitignore when you initialize a repo. There's no good reason to have to keep your app's node modules as well as your global installed modules on GitHqub. You can also throw in `.DS_Store`, it's an OSX file for the folder's icon, I believe. Either way, that should also always be in a .gitignore file as well. Awesome Job again.",

: "Hey Jeff, I cloned the app and tested it out. You've got the main functionality which is good. But there's no styling and the language should be cleaned up a little. If you want to use this later to show off as a portfolio or tech demo piece it'd be best to clean and style it up.\n\nThe main point that functionality works, now can you set it up and deploy to Heroku? It would only really be two if/else statements in your code. I think it's valuable practice and you should try to deploy after you tweak the front-end.\n\nAlso, when you init a git remo, don't forget to create a .gitignore file. There is no reason to be sending up your required node packages _and_ all your globally installed packages. Be lean, `npm install --save <packagename>`, don't bloat the app with other stuff we might not need. Also you can add `.DS_Store`to you `.gitignore` right away. It's just taking space and pixels on our screens. (it's a file related to how OSX displays icons... thus not needed on web, not even seen on Windows)\n\n",
: "Hey John, I cloned your repo and tried to spin it up locally, but had to run through a couple bugs to get it going.\n\nI haven't gotten to the Heroku deployment aspect, but I can see pretty early on that you're setting the node server port to 3000, which isn't available on Heroku (they don't want you choosing ports, so they have one for you). You can also write your code to run if deployed or local in a single line\n```\nvar PORT = process.env.PORT || 3000;\n```\n\nAlso in terms of deploying, your `config.js` is a bit confusing. First you declare your local connection and _then_ check if there's a deployment options using JawsDBURL? Then you use it if it's there else create a connetion to a remote server? You can take care of all these actions in one block of logic. e.g.\n```\nif (process.env.JAWSDB_URL) {\n connection = mysql.createCnnection(process.env.JAWSDB_URL);\n}else {\n connection = mysql.createConnection({\n   port: 3306,\n   host: 'localhost',\n   user: 'root',\n   password: '',\n   database: 'hamburger_db'\n   });\n};\n```\nAlso it's best not to keep sensitive information like logins and password on GitHub. You can use something like module.exports and requiring files that contain sensetive information. Just make sure you add those important to you `.gitignore`. Which is also missing, dont' forget when you initialize to add node_modules to the `.gitignore`, no sense in sending up all your globally installed modules _and_ possible settings files that contain private information.\n\nI also see remnants of the cats app where you didnt switch from `cats`to `hamburgers` which is breaking your app. There's also a simple spelling error on ou`/` route, which stops the app from moving forward, if i couldn't see the code.\n\nDid you get a chance to test this thoroughly on local? I see you tried to deploy to Heroku, but essentially the app wasn't even running for me on local. You should make sure it works locally first before trying to deploy.",

: "Hey Kevin, I checked out the app on Heroku and all functionality looks good to go. Nice work.\n\nI read through your code on GitHub, and it looks good to me. That's clearly seen as it works well as deployed.\n\nThe few things I noticed, we probably have talked about these in class though:\n- that .gitignore and the node_modules folder. probably just from the order of when you created and then `git add . ``'d -- next time just initialize with .gitinore and add node_modules to it before anything else\n- You've got lots of comments in you code, with the corresponding correct code right next to it. Have some faith in the code that works, it's nice to have notes, but it's double the reading for the same effect.\n- Flesh out the Readme.md: Give it an \"objective\", \"technologies used\", \"brief summary/takeaway\". It'll be useful for anyone who looks at the repo (possible employer), they can see you know ORMs, node, express, etc.   Not just like eating-burgers lol.\n- Maybe play around more with the css? Not necessary, just bored of looking at similar apps and colors. It would be a fun exercise now that you dont have to worry about the back-end giving your problems.\n\nOnce again, Awesome job. Keep it up.",
: "Nice Job Kelly, looks like you've nailed the functionality and the front-end is easy on the eyes. and responsive? +1!\n\nThere's not too much I can say it's a very well done homework and shows understanding of the topics for the week.\n\nThe few things that did catch my eye though, while not breaking would be the following:\n- a few missing semi-colons in your `burgers_controllers.js`\n- the Readme.md could use a better description (and a list of the technologies used)\n- in you local mysqlConnection `root:3000`? I am not aware of this, tried googling it, but it's late. Let me know what this does, if it's working it seems optional. \n\n- Also `.gitignore` file, initialize a repo with it or create one when you `git init`. You should almost aways by default add `node_modules` and `.DS_Store` before you start add/commiting anything. There's no reason why I need to see all your globally installed packages. I should only have to `npm install` and get the dependencies, nothing else needed.\n\nThe above are really minor issues, but worth looking at. The `.gitignore` to me at least is a must. You'll also use it store more valuable information per a project. I also don't want people putting valuable information up on GitHub. Anyway, great job again.\n",
: "Great Job Robert!",
: "Lisa you're sending the ssh path - can you submit a link to your github repo?\n",
: "great work on your custom orm!",
: "great work on the UX design as well!!!!",
: "great work on the hw. But in your controller.js file, instead of redirecting it to /burgers, i think it's just better to redirect it to '/'. Either way, it's not wrong just that it's kind of odd to be redirected to another route when you hit the home route.",
: "Keep up the good work! And i like the clean layout.",

: "Hey, it looks like you never got past setting up the file structure for this app (which looks good). You should finish it when you get the chance. I know you dont need the SQL practice, but the Express/Node stuff could be useful",
: "Hey David, the link you submitted doesnt seem to work. If `https://github.com/davidrips/burgers` is the correct repo, it looks like you didnt finish it. I think it would be really good for you to finish this, it may give you a better idea of MVC structure and allow you to get in some more practice with Express and Node.",
: "Hey, your commit message says the app isnt quite working, but other than having to change ```\n`date` TIMESTAMP ON UPDATE CURRENT_TIMESTAMP\n```\n to ````date` TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP```, the app seems to work just fine! Well done.",
: "Looks like you didnt get far with this one, you should go back and finish it if you find yourself with some of that \"free time\" I hear people talking about so much, especially if you need a node/express refresher.",
: "I was very tempted to dock you a grade for not staying true to the \"Eat-Da-Burger\" name, but given your artistic background, I'll trust the creative liberties you decide to take. Seriously though, amazing job. Nice touch with the burger history and the embedded gif too!",
: "I did not add my heroku link even though I attempted to deploy it there (https://secret-meadow-86040.herokuapp.com/). Ari mentioned on Slack that we are not required to upload this homework on Heroku. Just FYI. Thanks",
: "Keep up the great work, Caleb! This looks great!",
: "Great job on your homework. I have two suggestions though. One is to create the habit of creating a package.json file that lists your dependencies. And another one is to get rid of the files that are not being used in the project. :)",
: "Great job on your homework. Just a few tips... Avoid sending files that are not being used to Github. It's better to have a clean repository. :) Also, it would be better if you had the .DS file in .gitignore. And one last thing... Make sure to have \"https://\" before your heroku link when submitting your homework. ",
: "Hey Matt. So, a few things about your homework. First, on your server.js file, you passed waitingList to wait property instead of the variable that you created waitList. Then, inside your handlebars file, you referred to it by {{waitingList}}, but you should have referred to it as {{wait}} since that's the property that you set up. Then, I noticed that you named your handlebars file incorrectly. You added the html extension after handlebars, which you don't need to do. Finally, inside your handlebars, when you refer to {{id}} for the second time, you forgot to close the second }. Let me know if you need any helping fixing those errors and getting your app up and running. ",

: "Oh! And don't forget to add your node_modules to .gitignore. It shouldn't be on Github. ",

: "Hey Mary, you did the right homework. This is the one that Sean assigned to our class. So, a few things that you can do to fix your app. First, you required the tables Array and assigned it to the variable tables. So, you have to assign tables to a property of your choice and put in a object as the second argument of res.render... Like this: ```app.get('/tables', function(req,res) {\n   res.render('index', {table: tables});\n});``` . Then, inside your index.handlebars, you have to refer to that property in order to render something. In this case, you have an array. So, you have to do a `{{#each table}}`. And, in order to render each property inside that, you need to refer to it then. Like this : ```{{#each table}}\n\t\t\t\t\t\t\t<h2><span class=\"label label-primary\"> 1 | </span> {{customerName}}</h2>\n\t\t\t\t\t\t{{/each}}``` . Let me know if you want to see what I did to get your app working. Lastly, remember next time to add your node_modules to .gitignore before you add it to Github. ",

: "Great job on your homework! ",

: "Good job on your homework. Please double check if your application has been pushed correctly to heroku. Also, make sure that you don't push your node_modules to Github. Remember to add it to .gitignore. Lastly, if you are not requiring the data from a different file and you are not exporting those arrays out of server.js, you don't need to do ```module.exports = {\n\ttable: tableArray,\n\twaitlist: waitingArray\n}```. ",

: "Good job on your homework. Next time, make sure to follow the documentation to deploy a node.js app to heroku. https://devcenter.heroku.com/articles/deploying-nodejs. You don't need a php file and a composer.json to deploy node applications. Also, make sure that you don't send node_modules to Github. In addition, make sure that all your dependencies, everything you require, is installed with ```npm``` and ```--save```, like this ```npm install express-handlebars --save``` . The ```--save``` flag will save it automatically to your package.json file, which heroku will read through and install those dependencies in the server to run your app. If you don't have the node packages listed in package.json, heroku won't be able to install it and run important packages that you use on your app. ",

: "Good job on your homework. The only thing I want to add is that if you want to add the property of the waitingArray or tableArray inside your handlebars file, you can access it inside your #each, like this: ```{{#each tables}}\n\t\t\t\t\t\t\t<h2><span class=\"label label-primary\"> 1 | </span> {{customerName}}</h2>\n\t\t\t\t\t\t{{/each}} ```",

: "Great job on your homework. Make sure to push your application to heroku and give us the link to it. In order to push it to heroku, you need to edit your variable port to ```var port = process.env.PORT || 8080;```. Also, you need to have a start script inside your package.json that tells heroku to run ```node server.js```, or add a Procfile that runs your server.js file. Lastly, make sure to remain from sending node_modules to Github by adding it to .gitignore. ",

: "James, looks like you were off to a good start with rendering the handlebars file in the server.js file but you didn't complete the index.handlebars to each through the array and show each listing. Almost there! Take a look at the Ben and Jerry example on github to get an idea on what we were looking for here.",

: "Good job with this assignment! It all looks like it works as it should. Keep it up!",

: "Looks like you're missing the actual handlebars files and the routes that render them. I'm seeing the server and the engines but that's about it. Make sure you have a good understanding on how handlebars works by checking out the solution of other activities we've done in the past.",

: "I see you made the handlebar files correctly but you didn't create the route render calls. I know you have a good understanding on handlebars based on the last activity so just make sure you brush up if you have any questions by looking at other solutions for this type of assignment we've done in the past (mainly the BenAndJerry's example)",

: "Take a look at the folder structure [here](https://github.com/UCF-Coding-Boot-Camp/03-16-Class-Content/tree/master/0329-tth-class-content/14-express-deep-dive/14.2/5-BenAndJerrys). You need to make sure you follow this structure with the Views folder and the layout folder inside of that to get your handlebars files to be called correctly.We just finished project week so hopefully you have more experience with handlebars but if you're having issues, please reach out to us. ",

: "Dawn, take a look [here](https://github.com/UCF-Coding-Boot-Camp/03-16-Class-Content/tree/master/0329-tth-class-content/14-express-deep-dive/14.2/5-BenAndJerrys) and see how the structure works for handlebars. This includes calling the page in the route and passing the data. I'm grading this after project week and I know you've had more exposure to this. If you're still having issues, please come see us.",

: "Erik, take a look [here](https://github.com/UCF-Coding-Boot-Camp/03-16-Class-Content/tree/master/0329-tth-class-content/14-express-deep-dive/14.2/5-BenAndJerrys) to see how the structure and routes are set up. This is being graded after project week so hopefully you have more experience with handlebars at this point. If you're still having issues at this point with handlebars and the link above doesn't help, please come see us.",

: "This looks great. Nice job!",

: "What can I say? This is a fantastic effort. I like everything about the code, and the deployment to heroku looks like it went perfectly.",
: "Good start Aisha. It seems you are on the right track but got stuck at setting up your *model*.\n\nTry to setup a 1 on 1 so you can resolve any questions/concerns you have, because these underlying concepts are important to understand.\n\nOne thing I noticed that may help:\nYou did not put your content in an `app` folder, yet you reference it in your `server.js` file [here](https://github.com/aishadurrani/Lecker-Burger/blob/master/server.js#L12)",
: "Excellent work. Nice touch with the 'Visual Effects.'",
: "Good start Analben, but your heroku app did not work, and I could not get your code working on my local machine. It may be a good idea to meet after class and try to get working so you are clear on all the concepts.",
: "Great work Jacob. I noticed you did not use the MVC framework per the HW instructions. If you need to go over that again feel free to reach out.",
: "Excellent work Jeremiah! Good work tying all the concepts together from MVC, handlebars, sql, express, and node. Great job taking the UI to the next level with the re-order buttons and the layout.",
: "Good start on the HW. As you mentioned your ORM is not communicated with the DB. If you want we can sit down after class some day and take a look at this HW so that you can understand how to get ORMs and SQL working. ",
: "Great work Greg. Overall it seems you understand most of the main concepts. However I noticed that you didn't use the `burger.js` file (your Model). It is important you understand how to do this so it may be a good idea to meet with me or Peter to discuss you HW.",
: "Hi Faisal, I just couldn't understand how it fit in with this assignment. It seemed like an extra step that wasn't necessary. If you have the time I would definitely like to discuss it.\nThanks again!",
: "Excellent work! Happy to see you getting the hang of MVC and Handlebars. Also good work with your controller and setting up your routes.",
: "Good start, but it seems you didn't get devour part working. I didn't see a button for that so am not sure if you just didn't get to it or had issues with it. If you want we can take a look at it after class some time. \n\nHint: You seem to be getting some error with your ORM when you console log the `time` variable which seems to break the rest of your functionality and prevent anything from being added to the database.",
: "Hey Andrew, I wasn't able to test your app because you are missing the `keys.js` file. If you don't want to share the details can you slack me one with empty values. Thanks.",

: "Hey Faisal I just connected to the db w/o the keys.js file just to get it up and running.   It works perfectly, the code is clean, and well written. ",
: "Hey Jordan I got your app up and running and you got the front end all set up and you got your back end all set up, but they don't every connect. I opened up your burgers.js file and found out there wasn't any code in there. Was this a struggle to set up? Essentially, the intent of the burgers.js file is take in your data from your ORM and then create a constructor function out of it so it can be used by your controller to put data where it needs to be. \n\nBut what I did see is that you did directly connect your ORM to your api-routing.js, which is fine as long as it works, but for me it gave me an error stating that api/new does not exist. Did you get the same error? ",
: "Hey Kevin awesome job on the homework! Program runs smoothly and your code looks good! I only have one question why did you bring in:\n\nvar connection = require('./burger/config/connection.js');\nvar orm = require('./burger/config/orm.js');\n\nto your server.js Did it not work without it? Generally, you should need to import those to get your app up and running. If you need that code in your server.js file, there may be a deeper issue. \n",
: "Hey Lou I see you are having trouble. I think taking baby steps would be the best way to trouble shoot this issue. Can you successful get the data from the form to your database? You can figure this out by trying to successfully get your controller file to get data from your ORM -- those to need to work seemlessly for this app to function. ",
: "Hey Lou I cannot grade your homework because the way your repo is structured. I would delete your node modules file and re upload it. Its giving me errors when I try and clone it ",
: "Hey Mauricio can you remove your node modules folder from the repo it is giving me an error when I try to clone it. I think we went over creating a .gitignore file that file needs to be in any thing you make when you push it up to github. ",
: "Hey Nate love the app dude! You got a great eye for design and you got the app running smoothly. Your pretty well rounded already as a developer congrats!! \n\nI only have one suggestion you set up the routing of your app in your server.js file. For organizational purposes, I'd bring that could out and put it into a routing file and then import it into your server.js Its a mere suggestion to improve organization and to keep use to the concept of modular coding. ",
: "Hey Neena I couldn't get your app up and running. You gotta remember to add the .gitignore file so you don't upload your modules to the repo. It creates errors when other people try to clone your app. I'd try re uploading without the modules. ",

: "Hey Stephen I tried getting getting it up and running. You don't have a package.json file in your app. This will allow me to download the correct modules for this application. To get this going I would delete your node modules folder, type in npm init in terminal, run through the instructions, the re download your modules type in \"npm install 'name of package' --save\" and then all your module names will be saved to your package.json file and then I'll be able to download the required packages to run this app. ",
: "Hey Steven I noticed your app doesn't have any front end functionality handled by handlebars. I honestly need to brush up on this as well. Do you want to schedule a one on one to go over it together? ",
: "Hey Walt I got your app up and running but handlebars isn't configured properly. Do you want to go over after class on Monday. We can get handlebars up and running properly together. ",
: "Hey Zintis I have a good understanding of where you are what you know and what you don't know. One thing though I can show just remind me on slack is to set up a gitignore file so you don't upload your modules to github and give other people issues when they try to clone your app. Let's talk through slack to crystalize some of these concepts and to set up the gitignore file ",
: "Aaron, I'm trying to look at the HW, but I don't see a \"public\" folder where the \"index.html\" is being looked for. Did I miss something? Let me know if you need to upload something else.",
: "Hey Adam! This is some very nice work. I don't really have anything to correct; your files are broken up nicely so that each file has its own responsibility, and it looks like you got the submit button to work (your notes said you were going to try and fix it)\nThat said, if you have any questions about the exercise, please let me know!",
: "This was great! Great, great, great, GREAT! Awesome job. It was very easy to get right into your project and use your schema and seed file to get started.\n\nIndentation looks perfect, folder organization is on point. Overall amazing work!\n\nOnly suggestion -- You have your ACTUAL db credentials stored in your connection file. You may want to gitignore that so any jerks that may be crawling github don't give you a hard time.\n\nKeep up the awesome work!",
: "Hey Anthony -- It seems as if the routes are missing as well. Would you mind committing them to your repo so I can take a look and hopefully give you some insight into your handlebars problems? Thanks!",

: "Hey Bryant!\n\nWhen I loaded up your `server.js` file in node i got this error: `throw new TypeError('Router.use() requires middleware function but got a ' + gettype(fn));`.\n\nIn order to fix this you need to export your router. In your burger controller on line 5 I added this line: `module.exports = router;` and the server started.\n\nUpon loading, I was graced with this error:\n\n```\nTypeError: burger.findAllBurgers is not a function\n   at /Users/danvassallo/Desktop/sites/BurgerApp/controllers/burgers_controller.js:10:16\n   at Layer.handle [as handle_request] (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/layer.js:95:5)\n   at next (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/route.js:131:13)\n   at Route.dispatch (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/route.js:112:3)\n   at Layer.handle [as handle_request] (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/layer.js:95:5)\n   at /Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/index.js:277:22\n   at Function.process_params (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/index.js:330:12)\n   at next (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/index.js:271:10)\n   at methodOverride (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/method-override/index.js:65:14)\n   at Layer.handle [as handle_request] (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/layer.js:95:5)\n   at trim_prefix (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/index.js:312:13)\n   at /Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/index.js:280:7\n   at Function.process_params (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/index.js:330:12)\n   at next (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/index.js:271:10)\n   at urlencodedParser (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/body-parser/lib/types/urlencoded.js:88:40)\n   at Layer.handle [as handle_request] (/Users/danvassallo/Desktop/sites/BurgerApp/node_modules/express/lib/router/layer.js:95:5)\n```\n\nWhen I looked in your burger controller I noticed you had nested your model methods for your burger model in the routes for the burger. \n\nHad you created a burger model and set up its methods appropriately (Nearly identical to what you have going on in your routes) and required it like this `var burger = require('../models/burger.js');` I think you'd find your app operating a little better.\n\nI hope this clears some things up for you.",

: "Hey Chris! I've got your app started up and running no problem. Your orm makes reference to a `character` in your connection here `connection.query(s,[burger.name, character.name, character.role], function(err, result) {` and this is breaking your app -- Do you have a working version deployed to heroku or an updated schema file that would clear up these database inconsistencies?\n\nAlso this: \n\n```\nvar connection = mysql.createConnection({\n\tport: 3306,\n\thost: 'l9dwvv6j64hlhpul.cbetxkdyhwsb.us-east-1.rds.amazonaws.com',\n\tuser: 'yw8amioux4x36kvt',\n\tpassword: 'orh1zpd2u52t0cvp',\n\tdatabase: 'ta3fnb4w38no1l01'\n});\n```\n\nIs sensitive information. You may want to gitignore your connection file and maybe create a `connection.js.example` file that has the connection defined like this:\n\n```\nvar connection = mysql.createConnection({\n\tport: 3306,\n\thost: '*************',\n\tuser: '************',\n\tpassword: '***********',\n\tdatabase: '***********'\n});\n```\n\nNot that you've necessarily done anything wrong with this, just that someone could maliciously mess with your DB using these creds.",
: "Hey Dhaval!\n\nI had some trouble getting your app to start. It looks like there was an error in your SQL syntax on this line:\n`var s = 'SELECT * FROM' + tableInput + ';';`\n\nYou needed a space between your `FROM` and the following `'` like so:\n\n`var s = 'SELECT * FROM ' + tableInput + ';';`\n\nThanks for the schema file and the seed file! They were helpful although when your app started, I was simply greeted with `All of our data from the burger table` and there was no data populated.\n\nPerhaps reviewing how to iterate over database objects in handlebars will help you progress further should you decide to continue this project. Keep up the good work.",
: "I'd love to review your code but I'm getting a 404 on the repo : / Judging by the video, It seems you crushed this project. Find me in class and we can see about your heroku deployment issues.\n\nThanks for the video though, It looks like you're picking up the SQL stuff and really doing a great job with it!",
: "Deployed version works great Eugene! Awesome job! The folder organization in your repo is awesome as well!\n\nOnly thing worth mentioning in my opinion is that this:\n\n```\n\nvar connection = mysql.createConnection({\n   host   : 'l9dwvv6j64hlhpul.cbetxkdyhwsb.us-east-1.rds.amazonaws.com',\n   user   : 'q895fjvtwy61hg8g',\n   password : 'vnnx3icjdrw13eou',\n   database : 'imypcjlc2xdxdsh1'\n});\n```\n\nExposes sensitive information that you may not want out there. My suggestion is to create a `connection.js.example` file where these values are stubbed like so:\n\n```\n\nvar connection = mysql.createConnection({\n   host   : '***************',\n   user   : '***************',\n   password : '***************',\n   database : '***************'\n});\n```\n\nand then gitignore your actual connection file. Hope that helps you out in the future :)",

: "Hey James!\n\nI saw you have a `.gitignore` file in your repo -- add `node_modules` to it!\n\nYou also have 2 copies of your `seeds.sql` file in there, rm the one in the root and leave the one in the `db` folder.\n\nChange this line `var method      = require('method-override');` in your `server.js` to `var methodOverride      = require('method-override');`.\n\nYou are making reference to express-handlebars but you didn't run the `--save` flag when you installed it. Make sure to do this so it's added to your `package.json`.\n\nThis line `var routes = require('./controllers/burgers_controller.js');` in your server file throws an error because you have your controllers folder nested in your models folder. To fix this either move it outside of the models folder or change the line to this `var routes = require('./models/controllers/burgers_controller.js');`\n\nAlso this controller is empty, so this app isn't working all together : /\n\nIt looks like you had the right idea, but ran out of time. Keep up the good work!",
: "Jeremy,\n\nAmazing work on this project. Your code is clean and concise. Your folder structure is awesome and the production version works great!\n\nMy only recommendation is to create a `README.md ` file. This looks good to potential employers who may be looking through your repos and will also help future Jeremy figure out what past Jeremy did.\n\nCheckout this nice resource on git flavored markdown:\nhttps://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet\n\nKeep up the great work!",
: "Hey Kevin!\n\nGreat work as usual. First thing I noticed -- GITGNORE THAT .ENV FILE! I know from your repo that your JAWSDB url is `JAWSDB_URL='mysql://evmdgyboxmhl8h50:shfh247jpbyv7rb8@l9dwvv6j64hlhpul.cbetxkdyhwsb.us-east-1.rds.amazonaws.com:3306/r4xfclaqbyv9v6ag'` No, bueno!\n\nThat being said it's not that big of a deal -- you just don't want to do this in the future and expose REALLY sensitive stuff!\n\nLooking at the page, it seems like your css is a bit funny on the burger list (you probably want to set an `overflow: scroll`).\n\nOther than those slight tweaks -- this is great! Keep up the awesome work. \n\n\n",

<table>
<tr><td>: "Hey Leslie!\n\nLooks like your deploy is crashing! Womp womp. I've taken a look at your repo though. You need to create a `.gitignore` and add `node_modules` to it so we can trim the fat in this repo!\n\nYour routes aren't being found by the server file because your `app` folder name actually has a space in it and is only recognized with the escaped space like this `app\\ /` delete that space.\n\nYour api routes are empty as well. This is causing an error breaking the app. If you comment out the reference to them, the server starts.\n\nYour html routes are also broken. This `html-routes.js` should get you headed in the right direction:\n\n```\nvar express = require('express');\nvar app = express();\nvar path= require('path');\n\nmodule.exports=function(app){\n app.use(express.static('public'));\n\n\tapp.get('/', function(req, res){\n\t\tres.sendFile(path.join(__dirname, '../public', 'burger.html'));\n\t});\n\tapp.use(function(req,res){\n\t\tres.sendFile(path.join(__dirname, '../public', 'burger.html'));\n\t});\n}\n```\n\nKeep up the good work!\n",</td></tr>
<tr><td>: "Hey James,\n\nYou did the right thing by gitignoring your node_modules, but you didn't use the `--save` flag when running `npm install` on your node packages. Your project should include a `package.json` file with all of your node modules.\n\nYour app doesn't start because you aren't listening. Use this line in your server file to serve on your port: `app.listen(PORT);`\n\nOnce your server runs, it cannot GET '/' because your routing isn't set up correctly.",</td></tr>
<tr><td>: "Hey Raghav!\n\nFirst thing I would do is pull your project out of the nested `burger` folder and drop the contents in the root. Then, setup a `.gitignore` and `sudo rm -R` your `node_modules` folder. Your `package.json` will handle these packages so you don't need to commit them.\n\nMost of these files are near empty. The server file doesnt run because there is nothing there.",</td></tr>
<tr><td>: "Hey Robert!\n\nThe app looks and works great in production. I'd love to review your code though. Next time, send me your repo link as well!",</td></tr>
<tr><td>: " You did a great job, nice work ! If you have any other questions feel free to ask",</td></tr>
<tr><td>: "Hey Wojciech!\n\nA few notes --\n\nI would use `dot-env` to hide these creds:\n```\nvar connection = mysql.createConnection({\n   port: 3306,\n   host: 'l9dwvv6j64hlhpul.cbetxkdyhwsb.us-east-1.rds.amazonaws.com',\n   user: 'n9jogozvtox01zi0',\n   password: 'ewywzkkjw8759kj1',\n   database: 'w8eq483o13sjf0z8'\n});\n```\n\nI would change the port number from `80` to `8080`\n```\nvar PORT = process.env.PORT || 80; \n```\n\nChange this line:\n```\nvar s = 'SELECT id, burger_name, devoured, time FROM burgers ORDER BY time desc';\n```\n\nTo this because there is no time column in your schema:\n```\nvar s = 'SELECT id, burger_name, devoured FROM burgers ORDER BY time desc';\n```\n\nand your routes are a bit... funky. It looks like you've not required express. I'd look into that to assess the issues you are having with your orm performing these crud actions.",</td></tr>
</table>

: "Hey Shawn!\n\nWell, a few things were missing / didn't work but you were SO close.\n\nI changed this stuff to add a burger correctly:\n```\n#server.js\n\napp.post('/submit', function(req, res){\n\tconsole.log('adding a burger');\n\torm.addBurger(req.body.burger_name, function(data){\n\t\tres.redirect('/');\n\t});\n});\n```\n\n```\n#index.handlebars\n\n<form method=\"POST\" action=\"/submit\" class=\"button-size\">\n\t<textarea class=\"form-control\" type=\"text\" name=\"burger_name\"></textarea>\n\t<button type=\"submit\">Add it!</button>\n</form>\n```\n\nOther than that it looks good, you just need to get the devour route and action setup, and then pretty this thing up with a CSS framework! Good work!",
: "Hey Matt, everything in your app pretty much works except for a few kinks in how you specified the form submissions in your html. The method=\"\" and action=\"\" should correspond to the route you specified in your api-routing.js file. If you have any questions feel free to ask me",
: "Hey Maike!\n\nIt looks like you've committed your node_modules again. Create a `.gitignore` and add that folder to it.\n\nI tried to run your app. Your index.js file seems to be empty. I wasn't able to run the project at all.",